Subject: Re: Pid namespace patchsets review
Posted by ebiederm on Sun, 11 Mar 2007 01:57:13 GMT
View Forum Message <> Reply to Message

Herbert Poetzl <herbert@13thfloor.at> writes:

> IMHO not the best idea, mainly because both OpenVZ
> and Linux-VServer will end up either duplicating
> the pid code or using the incomplete (broken) version
> which probably gives the pid space a bad start ...
>
> I'd prefer to focus on fixing up the existing pid
> issues (conversion) first, then hitting it with a
> hopefully working pid namespace ...
>
> YMMV

Right now if we discount the kernel_thread to kthread conversion
we are probably 98% done with all of the conversions that make sense
without a pid namespace.

I guess NFS is the a big one still on the todo list.

The point is that there are only a handful of things that we know
about that we still need to convert that make a difference in
practice.

In addition the semantics of the pid namespace make a very big
difference in understanding how we need to group processes. Having
code people can look at an play with makes the subject a lot more
approachable.

Most of the remaining conversions do not actually make sense without
the pid namespace so we have work to do there.

Largely I am trying to structure this in a fashion that is accessible
to more people, which means more people can work on it together.

I think it would be reasonable to not merge the patch that enables
clone/unshare support upstream until we have everything else finished.

I have no intention of declaring a pid namespace done or complete
until it is but getting as close as we can get would be a real
advantage.

>> - When we do the rename can we please rename it task_proxy and have
>> the functions follow that naming. The resource limiting conversation

>> seems to be going in that direction, and it more general then what we
>> are using now.
>
> hmm, nsproxy was unusual but kind of understandable,
> task_proxy sounds just weird to me, I'd definitely
> prefer nsproxy over task_proxy, but I'm open for
> more 'space' related names too, like spaces or
> space_proxy or space_group ...
>

Well it is a proxy for task_struct and task_struct_proxy is just
long winded. Calling it task_proxy makes sticking the pointers
to other subsystems per task data more reasonable.

Eric

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers